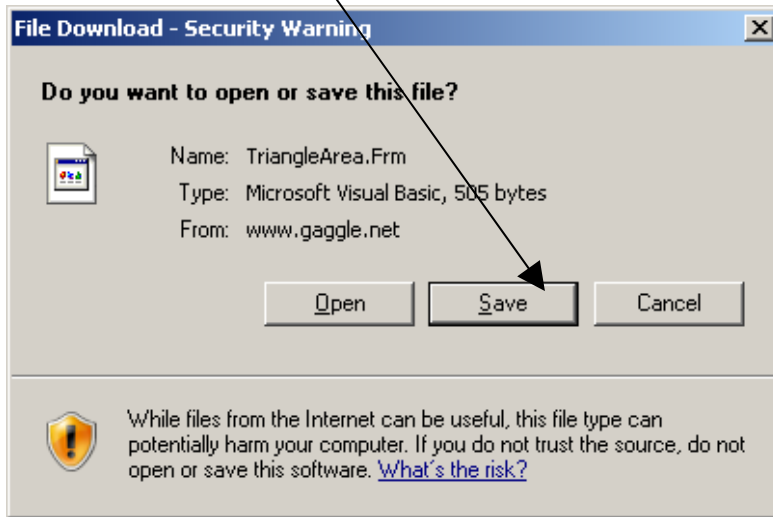


Code for TriangleArea.frm TriangleArea.vbp

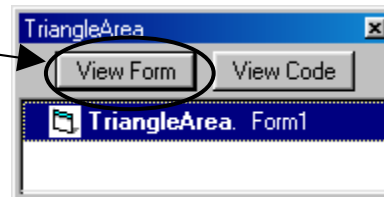


1. Double click on the Visual BASIC icon.
- 2.

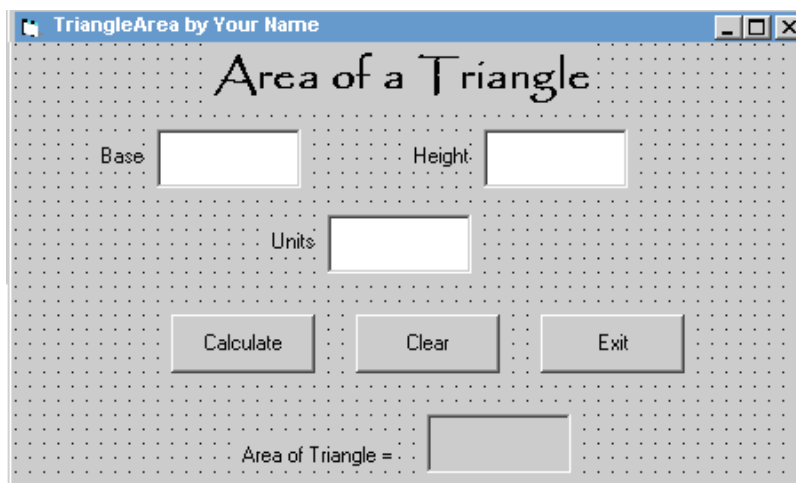
Find your TriangleArea.frm file in your Gaggle digital locker. Double click on it. When you see this pop up, choose **Save**, and save it to your **desktop** to continue working on it.



3. Click on the View Form button in the Project window.



4. Your form should look similar to this.



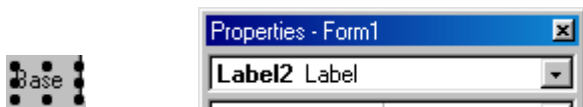
5. Make sure that all of your labels accept lblArea have the AutoSize property set to True.

Code for TriangleArea.frm TriangleArea.vbp

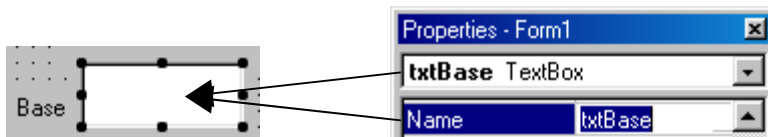
6. Click on the label that says **Area of a Triangle** at the top center of your form. Look at the top of the **Properties** window. It should say **Label1** Label. The word in **bold** is the name of the object. The word after that is the type of object it is. You do not need a special name for this label.



7. Select the label that says **Base**. In the **Properties** window, you see **Label2** Label. The word in **bold** is the name of the object. The word after that is the type of object it is. You do not need a special name for this label.



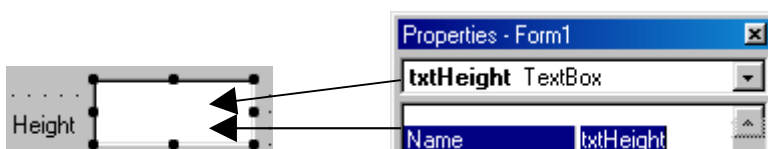
8. Click once on the white **TextBox** next to the word **Base**. In the **Properties** window, you should see **txtBase** TextBox. The word in **bold** is the name you gave the object. The word after that is the type of object it is. You need a specific name for this **TextBox** so that it can be referred to in the code of the program. In the code of the program it must be spelled exactly the same way as it is in the **Properties** window. Correct the **Name** property now, if it is not **txtBase**.



9. Select the label that says **Height**. In the **Properties** window, you see **Label3** Label. The word in **bold** is the name of the object. The word after that is the type of object it is. You do not need a special name for this label.



10. Click once on the white **TextBox** next to the word **Height**. In the **Properties** window, you should see **txtHeight** TextBox. The word in **bold** is the name you gave the object. The word after that is the type of object it is. You need a specific name for this **TextBox** so that it can be referred to in the code of the program. In the code of the program it must be spelled exactly the same way as it is in the **Properties** window. Correct the **Name** property now, if it is not **txtHeight**.

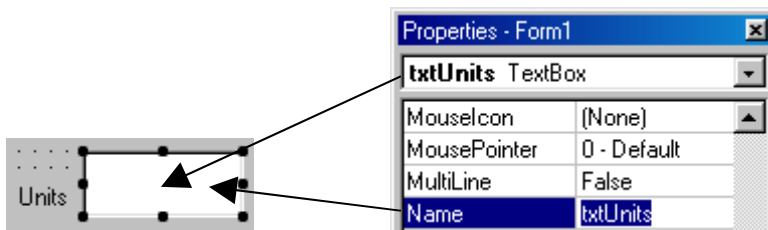


Code for TriangleArea.frm TriangleArea.vbp

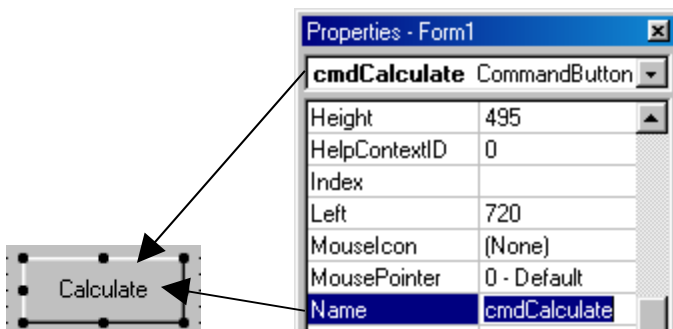
11. Select the label that says **Units**. In the **Properties** window, you see **Label4** **Label**. The word in **bold** is the name of the object. The word after that is the type of object it is. You do not need a special name for this label.




12. Click once on the white **TextBox** next to the word **Units**. In the **Properties** window, you should see **txtUnits** **TextBox**. The word in **bold** is the name you gave the object. The word after that is the type of object it is. You need a specific name for this **TextBox** so that it can be referred to in the code of the program. In the code of the program it must be spelled exactly the same way as it is in the **Properties** window. Correct the **Name** property now, if it is not **txtUnits**.



13. Click once on the **Calculate** **CommandButton**. Make sure the first **C** in **Calculate** is capital. In the **Properties** window, you should see **cmdCalculate** **CommandButton**. The word in **bold** is the name you gave the object. The word after that is the type of object it is. You need a specific name for this **CommandButton** so that it can be referred to in the code of the program. In the code of the program it must be spelled exactly the same way as it is in the **Properties** window. Correct the **Name** property now, if it is not **cmdCalculate**.

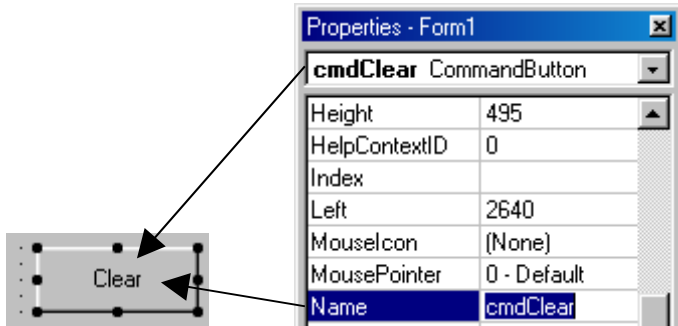


14. In the top toolbar click on the **Save** icon. 

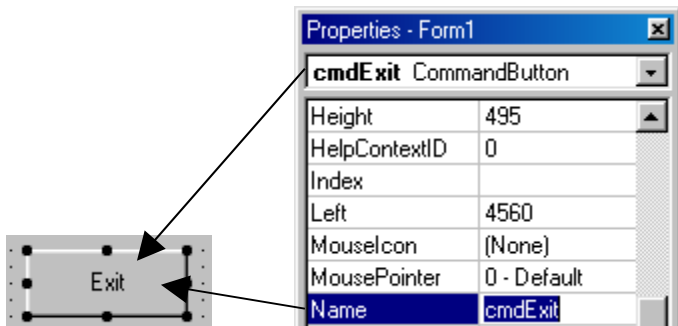
15. Click once on the **Clear** **CommandButton**. Make sure the first **C** in **Clear** is capital. In the **Properties** window, you should see **cmdClear** **CommandButton**. The word in **bold** is the name you gave the object. The word after that is the type of object it is. You need a specific name for this

Code for TriangleArea.frm TriangleArea.vbp

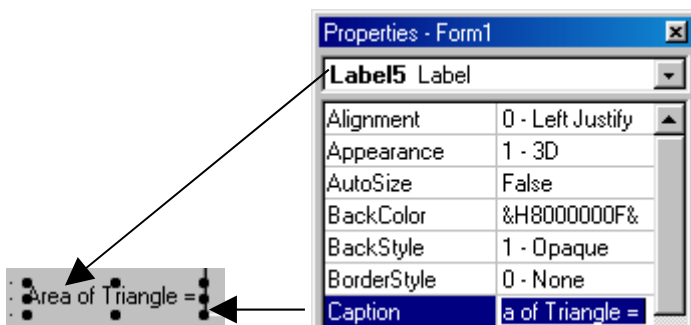
CommandButton so that it can be referred to in the code of the program. In the code of the program it must be spelled exactly the same way as it is in the Properties window. Correct the Name property now, if it is not **cmdClear**.



16. Click once on the Exit CommandButton. Make sure the first E in Exit is capital. In the Properties window, you should see **cmdExit** CommandButton. The word in **bold** is the name you gave the object. The word after that is the type of object it is. You need a specific name for this CommandButton so that it can be referred to in the code of the program. In the code of the program it must be spelled exactly the same way as it is in the Properties window. Correct the Name property now, if it is not **cmdExit**.



17. Select the label that says Area of Triangle =. In the Properties window, you see **Label5** Label. The word in **bold** is the name of the object. The word after that is the type of object it is. You do not need a special name for this label.



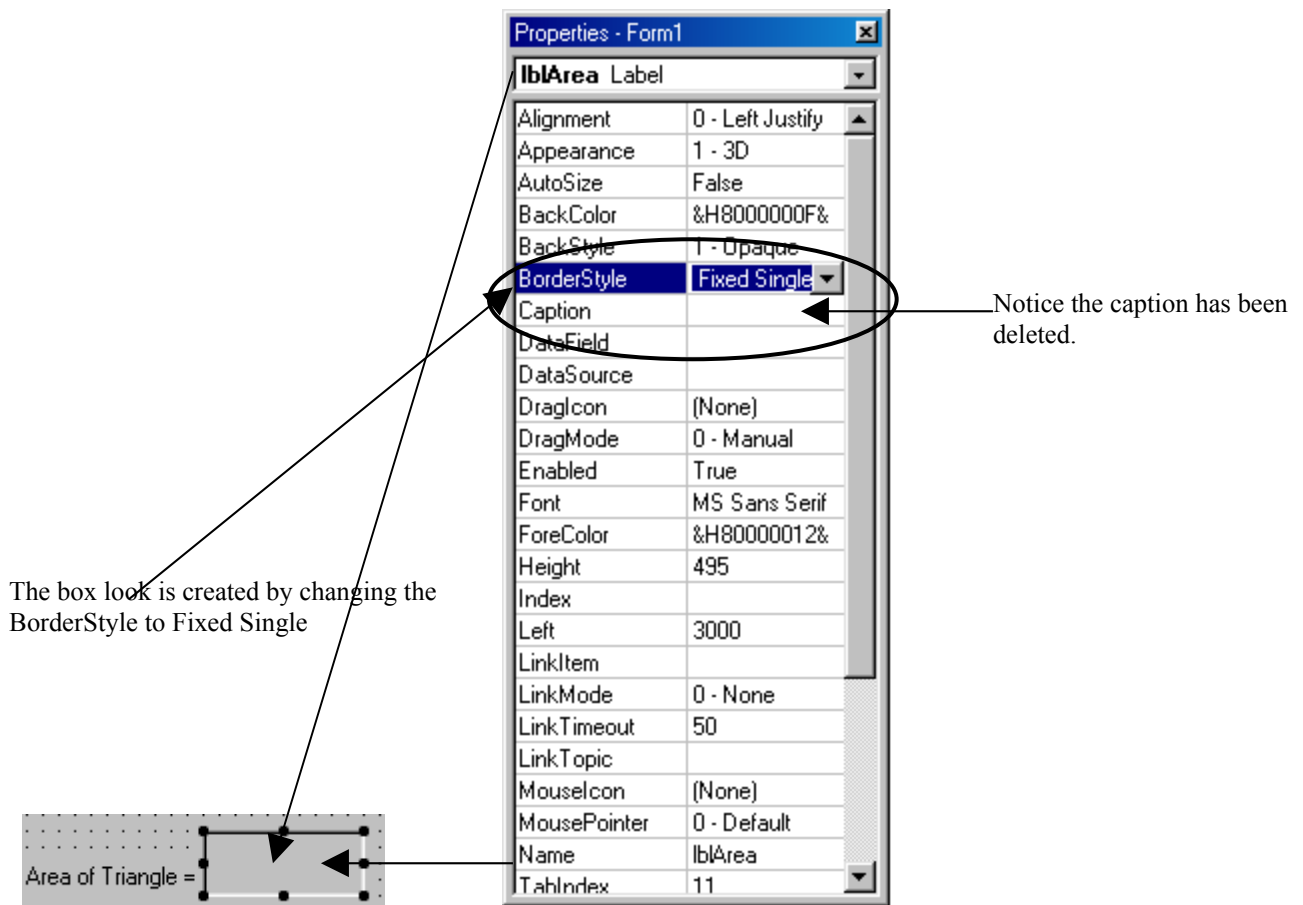
Code for TriangleArea.frm TriangleArea.vbp

18. Select the gray box to the right of Area of Triangle =. This is a Label with no Caption currently. It looks like a box because you changed the BorderStyle Property to Fixed Single. In the Properties window, you see **lblArea** Label. The word in **bold** is the name of the object. The word after that is the type of object it is.

You need a specific name for this Label so that it can be referred to in the code of the program. In the code of the program it must be spelled exactly the same way as it is in the Properties window.

Correct the Name property now, if it is not **lblArea**.

(lowercase **L** as in **l**ittle, lowercase **b** as in **b**oy, lowercase **A** as in **A**pple...)

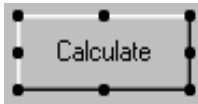


19. In the top toolbar click on the Save icon. 

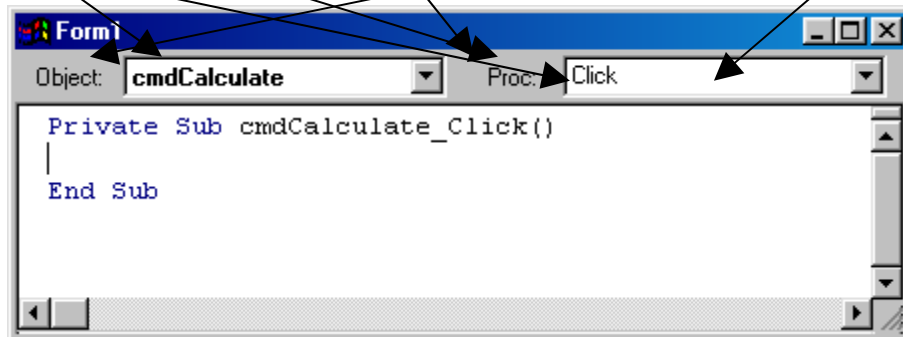
Now you are ready to use the BASIC programming language code to make the command buttons do what the caption indicates. Since Visual BASIC is event-driven, the code is placed in procedures for the event that is used to trigger the action, which is often the click event.

Compare all the parts of the following pictures with what appears on your computer monitor to make sure you are in the correct event procedure. Look at the words in the windows pictured.

Code for TriangleArea.frm TriangleArea.vbp



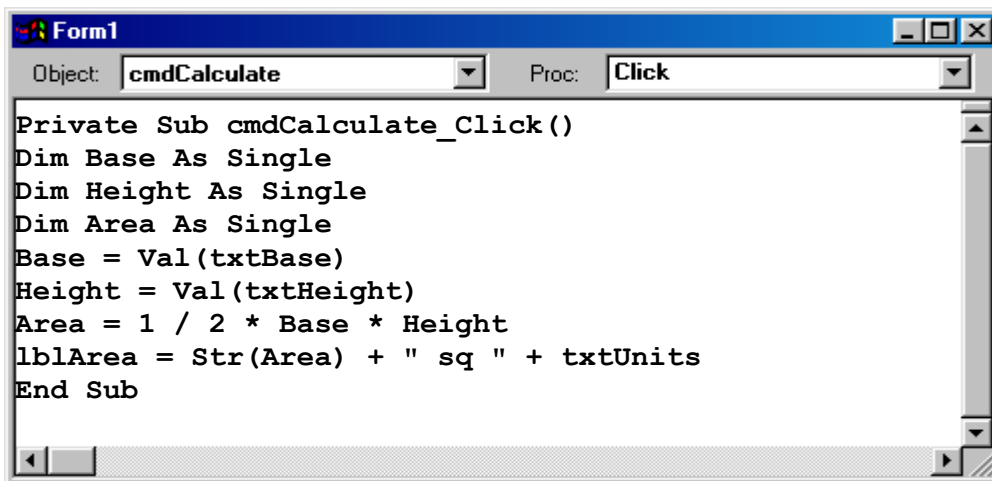
20. **Double click** on the Calculate CommandButton, so that the code window for the Object named **cmdCalculate** appears. The default Procedure for a command button is the Click event. The code in the Click Procedure is executed when the user **Clicks** on the Calculate command button.



21. Put the cursor under `Private Sub cmdCalculate_Click()` and type the following code:

```
Dim Base As Single
Dim Height As Single
Dim Area As Single
Base = Val(txtBase)
Height = Val(txtHeight)
Area = 1 / 2 * Base * Height
lblArea = Str(Area) + " sq " + txtUnits
```

It should look exactly like this.



22. Close the code window using the . In the top toolbar click on the **Save** icon.

Code for TriangleArea.frm TriangleArea.vbp

Q1: What are these statements for?

```
Dim Base As Single  
Dim Height As Single  
Dim Area As Single
```

A1: The lines that start with `Dim` are for telling the computer how much memory is necessary for storing the values of the variables.

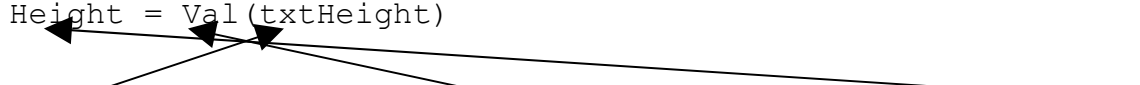
`Single` is the variable type for decimal numbers that are not excessively large and do not have a lot of digits after the decimal point.

Q2: What do these lines of code do?

```
Base = Val(txtBase)  
Height = Val(txtHeight)
```

A2: The default variable type for text typed into a `TextBox` is the `String` type which holds characters. Numerical calculations cannot be done with `String` values; therefore, these strings of numbers will be converted to actual numbers by using the `Val` function. `Val` is short for Value. In other words the *text* is converted to a *numerical value*. The numerical values are then stored in the variables that have been designated as decimal numbers with the `Single` variable type.

```
Base = Val(txtBase)  
Height = Val(txtHeight)
```



String (character) data converted to Single (decimal) data and stored in the appropriate variable.

Q3: What does this line of code do?

```
Area = 1 / 2 * Base * Height
```

A3: The formula for area of a triangle is $\text{Area} = \frac{1}{2} \text{base}(\text{height})$, but you must use an `*` for multiplication and a `/` for division. The variable that holds the answer must be on the left of the `=`.

Q4: What does this line of code do?

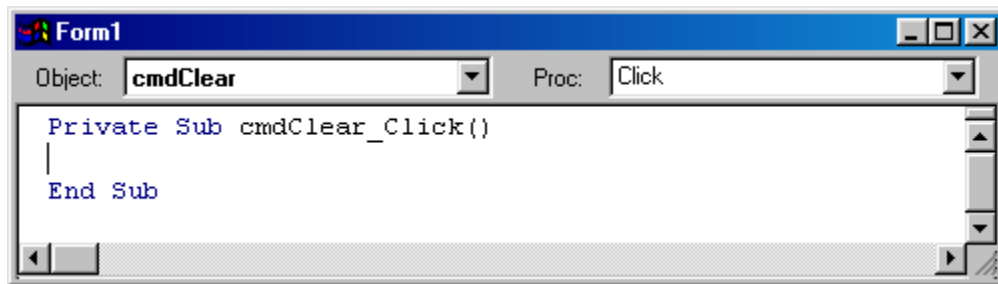
```
lblArea = Str(Area) + " sq " + txtUnits
```

A4: **lblArea** is the variable name for the label that will output the display of the answer. By default a label is the `String` variable type. Since `Area` is a `Single` (decimal number) it must be converted to a `String` with the function `Str`. The `+` symbol means to attach the character strings together. Quotation marks are needed to identify specific literal strings that are not given variable names.

Code for TriangleArea.frm TriangleArea.vbp



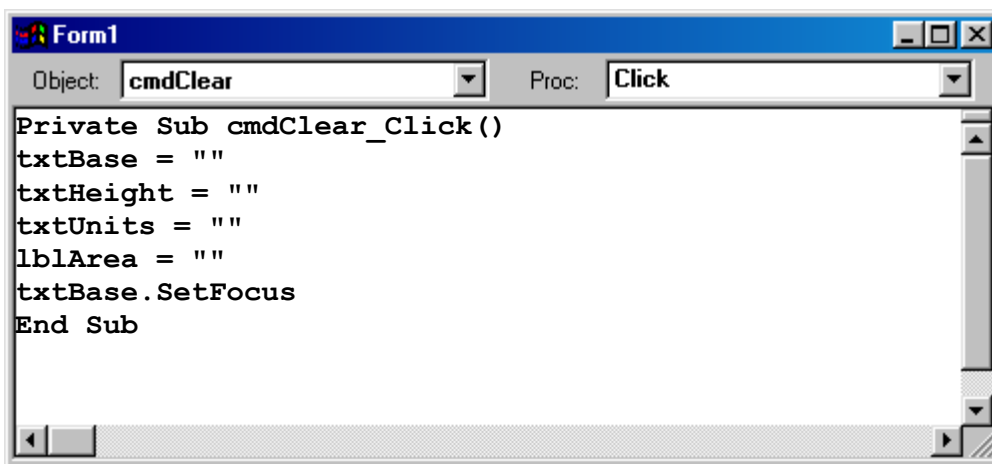
23. **Double click** on the Clear CommandButton, so that the code window for the Object named **cmdClear** appears. The default Procedure for a command button is the Click event. The code in the Click Procedure is executed when the user *Clicks* on the Clear command button.






24. Put the cursor under `Private Sub cmdClear_Click()` and type the following code:

```
txtBase = ""
txtHeight = ""
txtUnits = ""
lblArea = ""
txtBase.SetFocus
```

It should look exactly like this.




25. Close the code window using the . Click on the Save icon. 
26. Run the program by clicking on the Start icon  in the top toolbar.
27. Type 4.1 in the first Base textbox, then press Tab.
Type 9.5 in the Height textbox, then press Tab.
Type *cm* in the Units textbox, then click on the Calculate command button.

Code for TriangleArea.frm TriangleArea.vbp

You should see 19.475 sq cm in the gray label that looks like a box.

28. Click on the **Clear** command button. Notice that the textboxes and the lblArea label are now empty.
29. Try some other values. **10 20 ft** **100 sq ft**
30. Click on the **Exit** command button.
31. If you did not get the correct answer, look through the directions and through your program to find the errors, then correct them.

32. Click on the **Save** icon. 

Q5: What do these lines of code do?

```
txtBase = ""
txtHeight = ""
txtUnits = ""
lblArea = ""
```

A5: The textboxes and the answer label are assigned the `String` values with no characters in them. There is nothing between the quotation marks, so nothing will show in the boxes.

Q6: What does this line of code do?

```
txtBase.SetFocus
```

A6: It moves the cursor to the `txtBase` TextBox. What follows the dot after the TextBox name is the property that is being used. The `SetFocus` property puts the cursor in that TextBox.